SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

Coimbatore-35

DEPARTMENT OF BIOMEDICAL ENGINEERING

19BMB303 & Fundamentals of Microprocessors and Microcontrollers

Unit V - 32- BIT ARM PROCESSOR

III Year/ VI Sem

Dr. K. Manoharan,
ASP / BME / SNSCT

MICROCONTROLLER BASED SYSTEM DESIGN

Reduced Instruction Set Computer

Design Physiology

RISC Vs CISC Architecture

ARM Processor Architecture

ARM Core data flow model, Barrel Shifter

ARM processor modes and families

Pipelining

ARM instruction Set and its Programming

Pulse oximeter using ARM processor

Reduced Instruction Set Computer

Definition:
RISC stands for **Reduced Instruction Set Computer**.
It is a type of microprocessor architecture that uses a **small, highly optimized set of instructions**.

The main idea behind this is to simplify **hardware** by using an instruction set composed of a few basic steps for loading, evaluating, and storing operations just like a load command will load data, a store command will store the data.

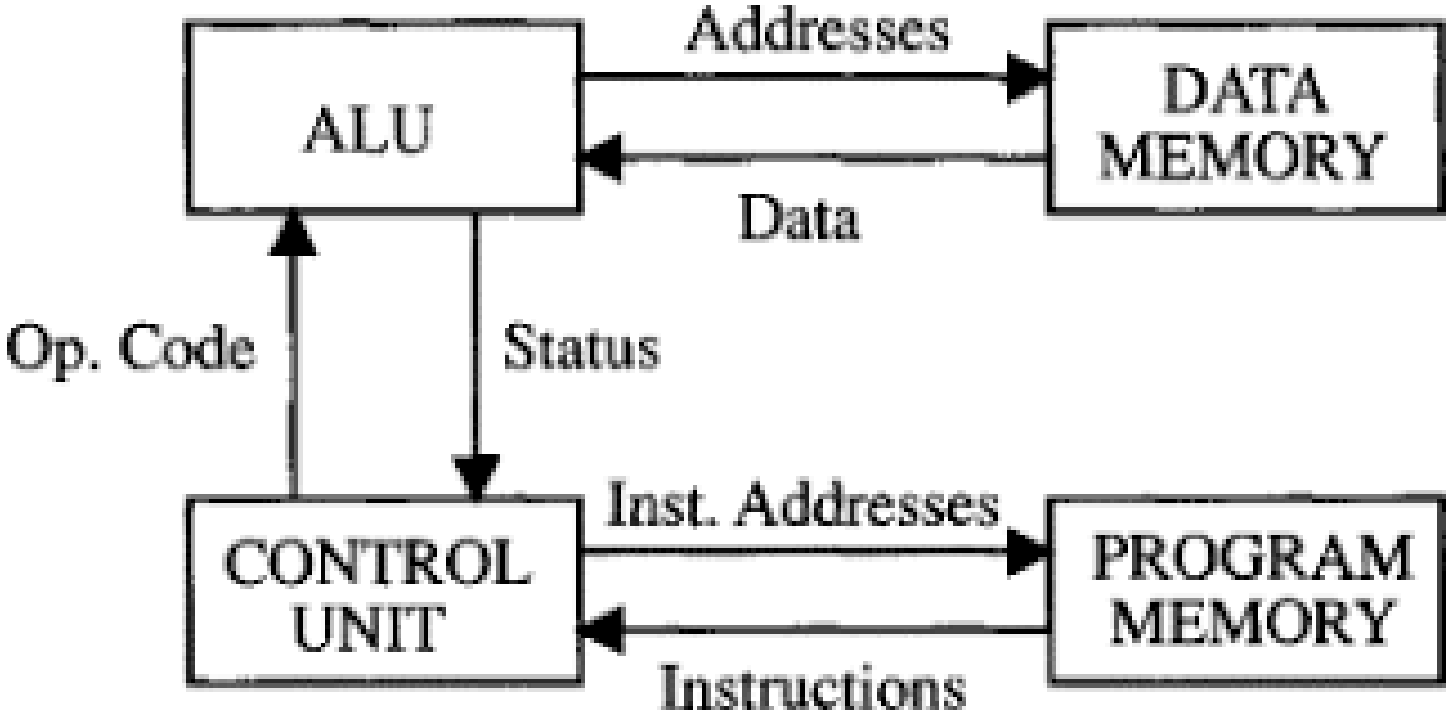
Goal:
Simplify instructions to **execute faster**

Reduced Instruction Set Computer

Key Features of RISC

- ✓ **Simple Instructions**:
  Each instruction performs a very small task.
- ✓ **Fixed Instruction Format**:
  Instructions are usually of **fixed length** and simpler to decode.
- ✓ **Load/Store Architecture**:
  Only **Load** and **Store** instructions access memory; all other operations are register-to-register.
- ✓ **Large Number of Registers**:
  Helps avoid memory access delays.
- ✓ **Pipelining**:
  Designed to use **instruction pipelining** efficiently.
- ✓ **Compiler Optimizations**:
  Compilers play a significant role in generating optimized code for RISC.

Reduced Instruction Set Computer



Reduced Instruction Set Computer
RISC Vs CISC

Feature	RISC	CISC
Instruction Size	Fixed	Variable
Execution Time	Mostly 1 cycle	Multiple cycles
Memory Access	Only Load/Store	Various instructions access memory
Hardware	Simpler	More complex
Code Size	Larger	Smaller
Example	ARM, MIPS, RISC-V	Intel x86, Motorola 68k

Instruction Pipeline in RISC

Typical RISC pipeline stages:

1. **Instruction Fetch (IF)**
2. **Instruction Decode (ID)**
3. **Execute (EX)**
4. **Memory Access (MEM)**
5. **Write Back (WB)**

- Pipelining allows overlapping of instructions, increasing throughput.

Reduced Instruction Set Computer

Advantages of RISC

- Faster execution (due to simple instructions).
- Easier to design and maintain.
- Better performance per watt (important for mobile devices).
- Easier to implement **instruction pipelines**.

Disadvantages of RISC

- Larger program size (more instructions needed for complex tasks).
- Greater burden on compiler to optimize code.
- Not ideal for systems with very limited memory.